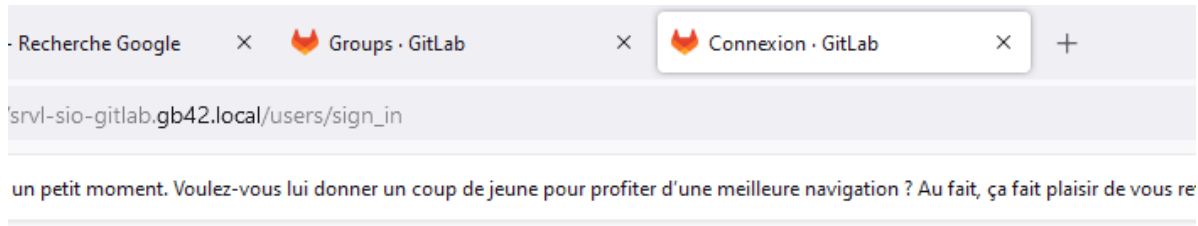


TP Gestion de version GitLab

GitLab est un outil de gestion de version de code source basé sur Git, une technologie de gestion de code source très populaire. L'objectif de ce TP est de créer, d'accéder à un dépôt distant de manière sécurisée ainsi que d'initialiser un projet.

I – Mise en place du dépôt distant

Dans un premier temps, nous devons nous connecter à notre compte GitLab sur le serveur 172.27.30.80.



GitLab Enterprise Edition

Nom d'utilisateur ou adresse de courriel principale

Mot de passe

[Mot de passe oublié ?](#)

☐ Se souvenir de moi

Connexion

Vous n'avez pas encore de compte ? [Inscrivez-vous maintenant](#)

Ensuite, nous devons créer un projet sur le serveur GitLab que nous allons nommer «WoodyCraft». Pour finir, nous créons un projet vide contenant un fichier README.txt.

II – Protocole SSH

Pour travailler sur les fichiers en local, il faut établir une connexion SSH. Le protocole SSH permet de communiquer de façon sécurisée.

Pour créer une clé SSH à insérer dans GitLab, nous devons aller dans profil>Paramètres utilisateurs>clés SSH.

Une clé SSH doit être renseignée dans le champ « Votre clé SSH ».

Pour générer une clé SSH, il faut ouvrir l'invite de commande puis utiliser la commande suivante : ssh-keygen -t ed25519 (ou ed25519 est un type de clé SSH).

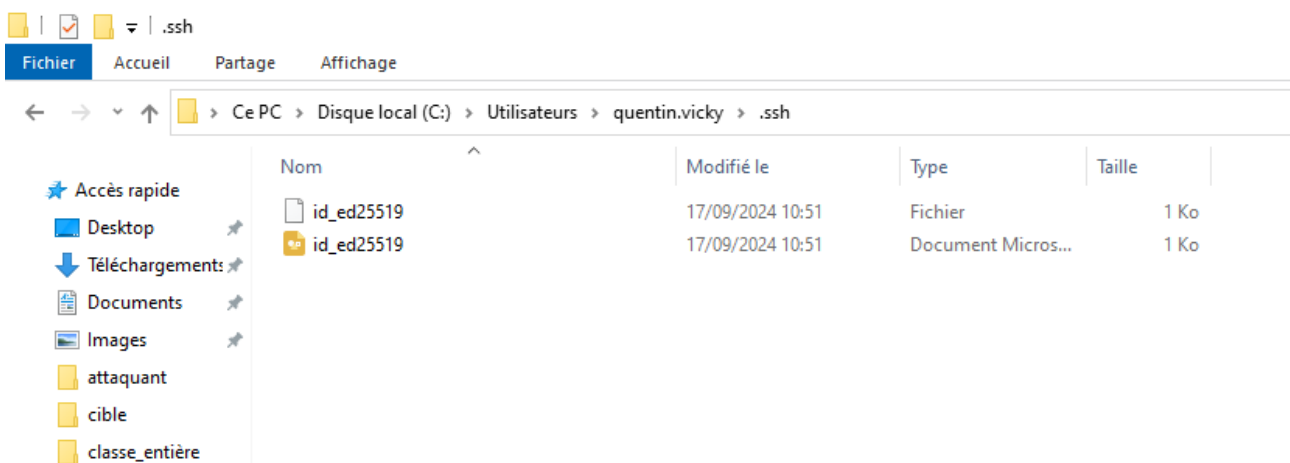
Voici une capture d'écran après avoir tapé la commande ssh-keygen -t ed25519 :

```
C:\Windows\system32\cmd.exe
Microsoft Windows [version 10.0.19044.2965]
(c) Microsoft Corporation. Tous droits réservés.

C:\Users\quentin.vicky>ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (C:\Users\quentin.vicky/.ssh/id_ed25519):
Created directory 'C:\Users\quentin.vicky/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\quentin.vicky/.ssh/id_ed25519.
Your public key has been saved in C:\Users\quentin.vicky/.ssh/id_ed25519.pub.
The key fingerprint is:
SHA256:+RaM3UQPMu0H9pvtCJ2UCx88wZ5vVU9ZU/lS5ikXW1k gb42\quentin.vicky@SI0-T10-07
The key's randomart image is:
+--[ED25519 256]--+
|      o.o. .B|
|      ++oooE|
|      o.=.**|
|      = oo @oO|
|      S + .* &*|
|      . ...B+=|
|      o  .o+|
|      .  . .|
+-----[SHA256]-----+

C:\Users\quentin.vicky>
```

On remarque que une nouvelle clé SSH s'est générée dans C:\Users\quentin.vicky/.ssh/id_ed25519 :



Il reste plus qu'à accéder au deuxième fichier contenant la clé publique SSH et de renseigner dans le champ clé de GitLab puis nous pourrons travailler en local sur les fichiers de notre projet hébergé sur GitLab.

III – Initialiser le projet

Nous devons maintenant créer un nouveau dépôt. Mais avant il faut configurer le global de Git avec les commandes suivantes à utiliser sur le terminal GitBash :

```
git config --global user.name «prenom nom »  
git config --global user.email «email»
```

Passons maintenant à la création d'un nouveau dépôt :

```
git clone git@srvl-sio-gitlab.gb42.local:woodycraft/woodycraftweb.git  
cd woodycraftweb  
git switch --create main  
touch README.md  
git add README.md  
git commit -m "add README"  
git push --set-upstream origin main
```

- On commence par cloner le répertoire en local
- On se déplace dans ce répertoire
- On crée une nouvelle branche locale « Main »
- On crée un fichier README.txt
- On pousse le fichier README.txt sur le serveur distant

Pour finir, nous devons créer une branche dev sur laquelle nous allons travailler durant les TP. Pour créer cette branche dev il faut taper les commandes suivantes. (Nous nous situons déjà dans le répertoire woodycraftweb) :

```
git branch dev ⇒ création d'une nouvelle branch « dev »  
git checkout dev ⇒ on se trouve maintenant sur cette branche
```

Nous avons créé la branche dev en local mais nous devons maintenant la mettre à jour pour qu'elle apparaisse sur le serveur distant. Pour cela, nous créons un fichier README.txt quelconque que nous allons pousser vers le serveur distant.

Voici les commandes utilisées :

```
touch README.txt  
git add README.txt  
git commit -m «ajout README.txt »  
git push --set-upstream origin dev
```

Conclusion :

Ce TP sur GitLab m'a permis de maîtriser les aspects essentiels de la gestion de version et de la configuration d'un environnement de développement :

J'ai appris à créer un projet sur GitLab, à configurer un dépôt distant, et à ajouter des fichiers comme le README.txt, me familiarisant ainsi avec l'interface et les fonctionnalités de GitLab.

J'ai compris l'importance de la sécurité dans les connexions entre dépôts locaux et distants en générant et en intégrant des clés SSH, garantissant des communications sécurisées.

J'ai cloner des dépôts, créer et gérer des branches locales, et à synchroniser les modifications avec le dépôt distant, notamment en créant une branche « dev » pour le développement.

La création, l'ajout, et le commit de fichiers, ainsi que leur transfert vers un dépôt distant, m'ont permis de maîtriser les commandes Git essentielles comme `git add`, `git commit`, et `git push`, pour maintenir un code source à jour et cohérent.